

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV MIKROELEKTRONIKY

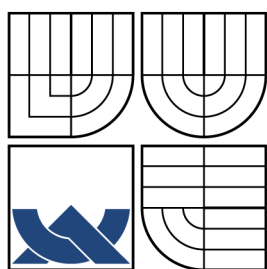
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF MICROELECTRONICS

IMPLEMENTACE ŘEZACÍHO MODULU
DO SOFTWAREVÉHO VYBAVENÍ TRIMOVACÍHO LASERU
AUREL ALS 300

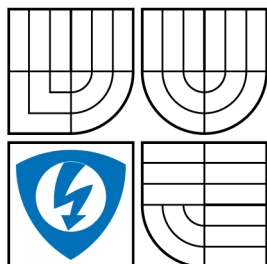
SEMESTRÁLNÍ PROJEKT
SEMESTRAL PROJECT

AUTOR PRÁCE
AUTHOR

MARTIN ŠTRYMPL



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV MIKROELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF MICROELECTRONICS

IMPLEMENTACE ŘEZACÍHO MODULU DO SOFTWAREVÉHO VYBAVENÍ TRIMOVACÍHO LASERU AUREL ALS 300 CUTTING MODULE IMPLEMENTATION INTO THE AUREL ALS 300 TRIMMING LASER SOFTWARE

SEMESTRÁLNÍ PROJEKT
SEMESTRAL PROJECT

AUTOR PRÁCE
AUTHOR

MARTIN ŠTRYMPL

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JAN PRÁŠEK, Ph.D.

BRNO 2011



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav mikroelektroniky

Semestrální práce

bakalářský studijní obor
Mikroelektronika a technologie

Student: Martin Štrympl

ID: 120617

Ročník: 3

Akademický rok: 2010/2011

NÁZEV TÉMATU:

**Implementace řezacího modulu do softwarového vybavení trimovacího laseru
Aurel ALS 300**

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s technologií pro realizaci hybridních integrovaných obvodů. Zaměřte se na oblast přesného nastavování hodnot tlustovrstvých rezistorů pomocí laseru Aurel ALS 300. Analyzujte jeho softwarové vybavení a modifikujte jeho zdrojový kód napsaný ve Visual Basic 6.0 tak, aby byl schopný řezat libovolné tvary načtené z některého z formátů třetích stran (PS, DWG, gerber). (Jedná se o kopplnění zadání celé závěrečné práce)

DOPORUČENÁ LITERATURA:

Dle doporučení vedoucího

Termín zadání: 20.9.2010

Termín odevzdání: 17.12.2010

Vedoucí práce: Ing. Jan Prášek, Ph.D.

prof. Ing. Radimír Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor semestrální práce nesmí při vytváření semestrální práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato práce se zabývá tvorbou řezacího modulu pro trimovací laser Aurel ALS 300. Konverzní program zajišťující konverzi formátu Gerber do vstupního formátu laseru byl napsán v jazyku C. Integrace konverzního programu do ovládacího programu laseru PROTOMAT byla provedena ve Visual Basicu 6.

Semestrální práce začíná úvodní teoretickou částí popisující základy tlustovrstvé technologie a užití trimování. Pokračuje analýzou vstupních formátů PostScript, DWG a Gerber. Jako vstupní formát pro konverzi je zvolen formát Gerber, který je podrobně analyzován. Analyzována jsou také vstupní data laseru Aurel ALS 300. Je popsáno, jakým způsobem byl vytvořen konverzní program, a jak s ním lze pracovat.

KLÍČOVÁ SLOVA

Tlustovrstvá technologie, hybridní integrovaný obvod, Gerber, trimovací laser, Aurel ALS 300, řezací modul, konverze

ABSTRACT

This project deals with the creation of a cutting module for trimming laser system Aurel ALS 300. The conversion programme designed to convert the Gerber format into the input laser format was written in C language. Integration of the conversion programme to the control programme of PROTOMAT laser was implemented in Visual Basic 6. The first part of the semester work describes the basic thick-film technology and the use of thick-film resistors trimming. An analysis of allowed input formats PostScript, DWG and Gerber follows. The Gerber format, which is analyzed in detail here, was selected as the input format for conversion. Input data of the laser Aurel ALS 300 are also analyzed. Finally the way of creation of the conversion programme and its use described here too.

KEYWORDS

Thick film technology, hybrid integrated circuit, Gerber, trimming laser, Aurel ALS 300, cutting module, conversion

ŠTRYMPL, Martin *Implementace řezacího modulu do softwarového vybavení trimovacího laseru Aurel ALS 300*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav mikroelektroniky, 2011. 51 s. Vedoucí práce byl Ing. Jan Prášek, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svůj semestrální projekt na téma „Implementace řezacího modulu do softwarového vybavení trimovacího laseru Aurel ALS 300“ jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedeného semestrálního projektu dále prohlašuji, že v souvislosti s vytvořením tohoto semestrálního projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu semestrálního projektu Ing. Janu Práškoví, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování projektu.

Brno

.....

(podpis autora)

OBSAH

Úvod	10
1 Technologie tlustých vrstev	11
1.1 Substráty	12
1.2 Sítotisk	13
1.3 Pasty a vrstvy	15
1.3.1 Vodivé vrstvy	15
1.3.2 Odporové vrstvy	16
1.3.3 Dielektrické a izolační vrstvy	16
1.4 Dostavování hodnot tlustovrstvých rezistorů	18
2 Volba zdrojového formátu	20
2.1 PostScript	21
2.2 DWG	23
2.3 Gerber	24
3 Konverzní program	26
3.1 Charakteristika Gerber formátu	27
3.1.1 RS-274X parametry	27
3.2 Analýza vstupních dat laseru	30
3.2.1 Datový soubor	30
3.2.2 Příkazový soubor	36
3.3 Generování výstupních dat	39
3.3.1 Generování výstupních dat v průběhu čtení vstupního souboru	39
3.3.2 Generování výstupních dat po dokončení čtení vstupního sou- boru	39
3.4 Charakteristika programu	40
3.4.1 Nemodulární provedení	40
3.4.2 Modulární provedení	40
3.4.3 Modul	40
3.4.4 Rozsah podpory Gerber formátů	44
3.4.5 Parametry	44
3.5 Integrace	46
4 Závěr	48
Literatura	49

SEZNAM OBRÁZKŮ

1.1	Síto a typické označení jeho vlastností	13
1.2	Princip sítotisku	14
1.3	Dva základní fyzikální principy využívané při sítotisku	14
1.4	Princip trimování hodnoty rezistoru laserem	19
2.1	Výsledek vykreslení příkladu PostScriptu	22
2.2	Gerber soubor převedený na obrázek	25
3.1	Návrh okna importu Gerber souboru	47

SEZNAM TABULEK

1.1	Typické vlastnosti substrátů pro tlustovrstvé hybridní integrované obvody	12
3.1	Parametry určující způsob zpracování souboru	27
3.2	Parametry obrazu	28
3.3	Parametry nástrojů	28
3.4	Parametry vrstvy	29
3.5	Různé parametry	29
3.6	Parametry modulu Bmp	42
3.7	Parametry modulu AurelALS300data	43
3.8	Parametry GerberConvertoru	45

LISTINGS

2.1	Ukázka PostScriptu	21
2.2	Ukázka Gerber souboru	24
3.1	Načítání příkazového souboru	30
3.2	Hlavička datového souboru	30
3.3	Načítání dat	32
3.4	Definice typu trimrecord	33
3.5	Typy řezu	36
3.6	Definice funkcí fGerber_Module_Start a fGerber_Module_End	40
3.7	Definice funkcí fGerber_Draw_LineSegment a fGerber_Draw_CircleSegment	41

ÚVOD

Tato práce se zabývá využitím trimovacího laseru Aurel ALS 300 (dále jen laseru) pro řezání obecných tvarů v rovině. Trimovací laser je primárně určen pro dostavování hodnot tlustovrstvých rezistorů v hybridních integrovaných obvodech (HIO) vyráběných technologií tlustých vrstev. Laser je ovládán programem PROTOMAT. Informace o jednotlivých řezech získává PROTOMAT ze vstupních souborů, jejichž formát je programu přizpůsoben. Data o řezech generovaná externím programem (např. AutoCAD) však nejsou ve formátu čitelném programem PROTOMAT. Přípustným formátem externích dat je PostScript, DWG nebo Gerber. Pro konverzi souboru v jednom z uvedených formátů do formátu čitelného programem PROTOMAT bude v rámci této práce vyvinut konverzní program. Program PROTOMAT bude modifikován přidáním grafického rozhraní pro uživatelsky příjemný import dat z externím programem generovaného souboru.

Počátky rozvoje tlustovrstvé technologie se datují do roku 1942. Okolo roku 1950 bylo zjištěno, že tlustovrstvou technologií lze vytvářet desky plošných spojů. V 60. letech 20. století vznikly odporové pasty na bázi stříbra, uhlíku a oxidů mědi, ale jejich elektrické vlastnosti byly nevyhovující, a proto byly vyvinuty odporové pasty na bázi paladia, čímž se elektrické vlastnosti podstatně zlepšily. To byl jeden ze základních kroků směřujících ke vzniku HIO. [1]

Při výrobě HIO technologií tlustých vrstev se na vhodný substrát nanáší pasty, které po výpalu plní určitou funkci. Základní pasty jsou vodivé, odporové, dielektrické a izolační. Nanášení past probíhá obvykle přes síto s motivem. Postupnou sekvencí nanášení a vypalování se vytváří výsledný HIO.

Odporové pasty jsou charakterizovány hodnotou odporu na čtverec a teplotní závislostí odporu. Konečný odpor tlustovrstvého rezistoru se odvíjí od počtu čtverců a použité odporové pasty. Rozptyl odporu rezistorů po výpalu je $\pm(15 \text{ až } 20\%)$ a není vždy nutné použít dostavení, i když vrstvy mohou být opracovány laserem YAG : Nd [2]. Laser umožňuje dostavení libovolného rezistoru na žádanou hodnotu s přesností vyšší než 1%. Kupříkladu firma HC electronics udává možnost nastavení libovolných hodnot rezistorů s přesností až 0.2% [3].

1 TECHNOLOGIE TLUSTÝCH VRSTEV

Tlustovrstvé hybridní integrované obvody jsou vytvářeny tiskem a výpalem past na nosné korundové podložce ve tvaru obrazců rezistorů a kondenzátorů propojených vodivými spoji, jsou doplněné vsazovacími pasívními i aktivními součástkami a jako celek jsou povrchově chráněny plastovým, keramickým nebo kovovým pouzdem [2]. Přednosti tlustovrstvých HIO jsou v technickém možnostech technologie. Oproti technologii tenkých vrstev je technologie tlustých vrstev velice flexibilní. Tlustovrstvé HIO také dosahují vyšší spolehlivosti než ekvivalentní obvody vytvořené na desce plošných spojů (DPS) z diskretních součástek. Při přechodu obvodu sestaveného na DPS z klasických součástek do tlustovrstvé technologie se často ušetří místo, objem i hmotnost a zvýší se spolehlivost. Současná materiálová základna umožňuje vytvářet pasívní vodivé, odporové i dielektrické vrstvy ve velmi úzkých tolerancích a s parametry, které mnohdy předčí parametry klasických součástek, u vsazovacích součástek je navíc možnost předběžného výběru [2].

Technologie tlustých vrstev je vhodná i pro malosériovou výrobu. Dodací lhůty mohou být krátké při malých investičních nákladech. Výhoda použití technologie tlustých vrstev se může projevit již v sériích od 100 kusů obvodů, dle složitosti. Toho nelze dosáhnout u žádné jiné technologie výroby IO.

Základní body výroby HIO technologií tlustých vrstev podle [2]:

- návrh zapojení hybridního integrovaného obvodu,
- návrh topologie,
- zhotovení sítotiskových šablon,
- sítotisk a výpal,
 - podložka,
 - pasta vodivá, odporová, dielektrická,
- dostavení pasívních součástek,
- kontrola pasivní sítě,
- montáž,
 - vsazovací součástky,
- pájení vývodů,
- oživení a nastavení obvodu,
- pouzdření,
- elektrická kontrola,
- expedice.

V roce 1980, tedy v době největšího rozmachu tlustovrstvé technologie, bylo tlustovrstvou technologií vyráběno 60 % až 80 % hybridních integrovaných obvodů hlavně z důvodů velké flexibility technologie, nízkých investičních nákladů a hospodárnosti výroby.

1.1 Substráty

Substrát HIO bývá nejčastěji z korundové keramiky. Je možné se setkat i s keramikou berylnatou, titanátovou (titanát barya), steatitovou (steatit), forsteritovou (forsterit) či zirkonovou stejně jako se sodno-vápenatými či borosilikátovými skly. Nejrozšířenějším materiálem je korund s obsahem 96 % Al_2O_3 , ke kterému jsou dilatačně přizpůsobeny vodivé, odporové i dielektrické pasty [2]. V tabulce 1.1 jsou vypsány některé vlastnosti korundové a berylnaté keramiky.

Berylnatá keramika se používá v silně tepelně namáhaných obvodech. Zde se uplatňuje její vysoká tepelná vodivost. Výhodou je podobný součinitel lineární teplotní roztažnosti jako u korundové keramiky.

Tab. 1.1: Typické vlastnosti substrátů pro tlustovrstvé hybridní integrované obvody [2]

Parametr	Jednotka	Korundová keramika 96 % Al_2O_3	Berylnatá keramika 99.5 % BeO
Drsnost povrchu R_a	μm	0.5 až 1.3	0.5 až 1.3
Měrná tepelná vodivost λ při 25°C	$\text{W.m}^{-1}.\text{K}^{-1}$	18 až 36	240
Součinitel lineární teplotní roztažnosti α (25 až 300°C)	K^{-1}	6.4×10^{-6}	6.5×10^{-6}
Elektrická pevnost E_d	kV.mm^{-1}	8.3	8.7
Měrný odpor ϱ při 300°C	Ωcm	10^{10}	10^{12}
Měrný odpor ϱ při 500°C	Ωcm	10^7	10^{11}
Relativní permitivita ε_r (1 MHz)	—	9	7
Ztrátový činitel $\tan \delta$ (1 MHz)	—	10^{-4}	10^{-3}
Měrná hmotnost	g.cm^{-3}	3.7	2.9
Specifické teplo	$\text{J.kg}^{-1}.\text{K}^{-1}$	878	1546
Maximální použitelná teplota	$^\circ\text{C}$	1550	1500
Pevnost v ohybu	MPa	320	170

1.2 Sítotisk

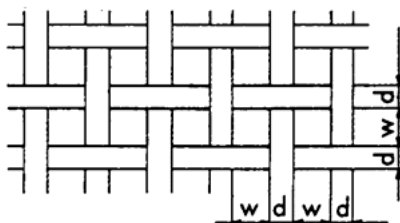
Cílem sítotisku je vytvořit z pasty na substrátu obrazec žádaného geometrického tvaru a tloušťky. Vytvoření žádaného obrazce lze dosáhnout i jinými technikami (stříkáním, máčením, leptáním, razítkováním apod.). Sítotisk je však nejlépe přizpůsobitelný potřebám HIO. Principem sítotisku je protlačování pasty skrze volné otvory síta. Kromě sítotisku se pro nanášení tlustých vrstev používá také šablonový tisk a popis.

Základní výrobní prostředky pro sítotisk jsou sítotiskový strojek s uchycením pro rámeček nesoucí síťovou šablonu (na ní je vytvořen požadovaný motiv - maska) a s držákem pro uchycení keramického substrátu (substrát je vakuově přidržen a fixován v jedné poloze), dále stěrka na protlačení sítotiskové pasty (může být jak ruční, tak vedená mechanicky jako součást strojeku) a konečně sítotisková pasta. [4]

Vlastnosti síťoviny (tkaniny) určuje průměr jejího vlákna (d), světlost oka (w) a tloušťka (D). Z těchto údajů se určuje hustota tkaniny (n) a světlost síta (A_0). Pro výpočet hustoty tkaniny lze použít rovnici 1.1 a pro výpočet světlosti síta rovnici 1.2. Znázornění průměru vlákna a světlosti oka lze nalézt na obrázku 1.1.

$$n = \frac{10 \text{ mm}}{w + d} \quad (1.1)$$

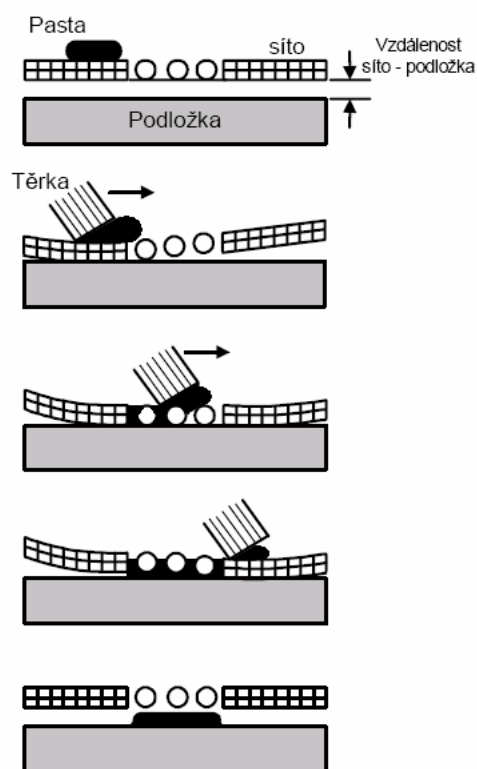
$$A_0 = \frac{w^2}{(w + d)^2} \times 100\% \quad (1.2)$$



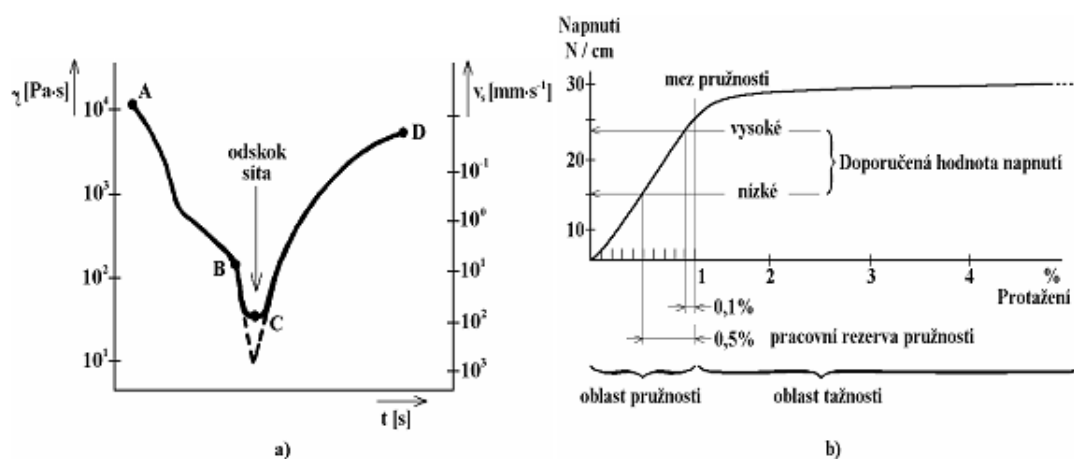
Obr. 1.1: Síto a typické označení jeho vlastností [5]

Pro sítotisk je nutné vytvořit sítotiskové šablony. Ty se vytvářejí přímo na síťovině nebo nepřímo pomocí kovové či plastové fólie, která se na síto přikládá.

V době tisku musí být síto vypnuto ve vzdálenosti nazývané odtrh od substrátu. Díky tlaku stěrky se síto protahuje (obr. 1.2 a obr. 1.3b) a současně dochází ke změně viskozity sítotiskové pasty (viz. obrázek 1.3a), která protéká skrze síto na substrát, kde ulpívá.



Obr. 1.2: Princip sítotisku [5]



Obr. 1.3: Dva základní fyzikální principy využívané při sítotisku - a) závislost viskozity pasty na tlaku; b) závislost napnutí síta na jeho protažení [6]

1.3 Pasty a vrstvy

Materiály pro tlusté vrstvy jsou dodávány ve formě past [4]. Viskozita past se pohybuje v rozsahu 50 až 80 Pa.s. Pasty jsou tvořeny základními složkami, které nazýváme funkční, vazební (tavivové) a lakové (pojivové).

Funkční složka určuje druh pasty. Například v případě vodivých past může jít o částice drahých kovů. Velikost částic ve složce musí být oproti velikosti ok síta dostatečně malá. U dielektrických a izolačních past se používají dielektrické materiály či skelné frity. V případě odporových materiálů je složení závislé na použitém odporovém systému. Základem mohou být směsi drahých kovů či oxidy.

Tavidlová složka je tvořena skleněnou fritou, jejímž posláním je vytvoření vazby mezi funkční složkou a substrátem. Proto se používají pro tavivovou složku nízkotavná skla s teplotou měknutí již od 600 °C. V poslední době se používají stále více pasty s oxidovou vazbou vytvořenou přídavným kovem (např. 4 % Cu do Au). Při výpalu musí dojít k měknutí, nikoli však k roztavení, aby vznikla nosná matrice pro funkční složku. [4]

Laková složka se v průběhu výpalu pasty odpařuje. Nemá tedy vliv na konečné vlastnosti pasty. Ovlivňuje však vlastnosti nevypálené pasty, a tím ovlivňuje i proces sítotisku. Laková složka je tvořena organickými látkami (např. terpeneol, butyldiglykolacetát).

Sítotiskové pasty dělíme na vodivé, odporové, dielektrické a izolační.

1.3.1 Vodivé vrstvy

Jako hlavní složky pro vodivé vrstvy se používají především drahé kovy: stříbro, zlato, paladium a jejich slitiny z důvodu stálosti a tepelné odolnosti, dále pak kovy obecné jako hliník, nikl, měď, které jsou cenově výhodnější [2]. Při přepočtu ceny na plošný odpor vodiče, nejsou mezi materiály zřetelné ekonomické rozdíly.

Stříbro je poměrně levný materiál, který má vynikající pájitelnost a elektrickou vodivost (elektrody keramických kondenzátorů, stínění apod.) [2]. Použití stříbra v tlustých vrstvách je omezeno rychlou rozpustností v pájce Sn-Pb a snadnou migrací. Rozpustnost i migraci lze snížit přidáním paladia (asi 25 %).

Pro tlustovrstvé vodiče jsou rozšířeny pasty na bázi Ag-Pd. Mají velkou vodivost a vynikající pájitelnost. Také se nechají snadno slučovat s mnoha odporovými a dielektrickými pastami. Společný výpal kontaktní a odporové pasty při teplotě nižší než 800 °C může zhoršit pájitelnost vlivem přítomného PbO [2]. Alternativou pastám na bázi Ag-Pd mohou být pasty na bázi Ag-Pt.

Zlaté pasty se skleněnou fritou jsou používány pro eutektické připojení čipů polovodičů a vodičů termokompresním nebo ultrazvukovým svařováním [2]. V pájce

Sn-Pb se rychle rozpouští. Pasty založené na chemickém spojení zlata s keramikou podložkou bez skleněné frity mají lepší přilnavost i po opakovaném výpalu bez degradace adheze [2]. Po výpalu vrstva obsahuje až 99 % kovu. Hloubka vniku do podložky je velmi malá, což je výhodné pro mikrovlnné aplikace.

Dále je možné se setkat s pastami na bázi Au-Pb, které jsou odolnější proti rozpouštění v pájce Sn-Pb, pastami na bázi mědi, niklu či těžko pájitelných kovů. Vodivé vrstvy na bázi mědi se vyznačují velkou vodivostí, vynikající pájitelností a dobrou odolností proti rozpouštění v pájce [2]. Zcela zvláštní skupinu tvoří foto-rezistivní vodivé cesty, které polymerují po ozáření ultrafialovým světlem.

1.3.2 Odporové vrstvy

Od rezistorů vytvořených v technologii tlustých vrstev jsou očekávány stálost, dobré elektrické vlastnosti, malé rozměry a hospodárnost výroby. Výchozím materiálem je odporová pasta složená ze směsi kovů nebo slitin (Au, Ag, Pt, Pd, Ag-Pd), polovodičových kyslíčků kovů (Pb, In, Tl, Ru, Ir, Rh, W, Os) a skleněné frity (borosilikátové nebo hlinitosilikátové sklo) [2]. Pro lepší tiskové vlastnosti se přidávají smáčecí prostředky. Tloušťka natisknuté vrstvy se pohybuje mezi 20 a 30 μm . Po natisknutí se 5 minut ponechává pasta volně na vzduchu, aby došlo k slití stop po sítu. Následuje sušení při 125 °C po dobu 10 minut.

Prášky kovového paladia a stříbra smíchané s borosilikáto-olovnatým sklem tvoří pasty na bázi Pb-Ag. Poměrem kovů a skloviny se mění plošný odpor vrstvy, teplotní součinitel odporu, proudový šum a stabilita vrstvy.

Odporové vrstvy Ru-sklo používají ruthenia buď kovového, nebo jako kyslíčků [2]. Výhodou vrstev RuO_2 je velmi nízká citlivost na teplotu vypalování.

1.3.3 Dielektrické a izolační vrstvy

Dielektrické pasty se používají jednak pro vytváření tlustovrstvých kondenzátorů, izolaci křižujících vodičů, vytváření vícevrstevných struktur, ochranu před vlivem prostředí, těsnění a pouzdrění. Pasta je tvořena směsí skloviny, keramického materiálu a organického nosiče.

Pro izolaci dvou křižujících se vodičů lze použít i jednoduché skloviny. Skloviny se používají pro některé méně náročné aplikace i přes své nedostatky. Rekrytalizační sklovina složená ze skla a devitrifikačních přímísenin nalézá daleko větší použití. Smíšením oxidu hlinitého se sklem dostaneme sklovinu s nulovou tekutostí, kombinací skla a keramiky.

U kondenzátorů tvoří dielektrická vrstva funkční část součástky. Zde je kladen důraz na relativní permitivitu, ztrátový činitel, izolační odpor, teplotní a kmitoč-

ovou charakteristiku a stabilitu. Používané materiály se připravují z práškových feromagnetických materiálů a skla. Výběr dielektrika kondenzátoru vymezuje frekvenční oblast použití. Rozhodující je ztrátový činitel.

U některých vrstev je nutné aplikovat vrstvu ochrannou, která chrání vrstvy proti vlivům prostředí (atmosféry apod.). Ochranné vrstvy jsou tvořeny jednoduchými sklovinami se součinitelem teplotní roztažnosti $7.5 \times 10^{-6} K^{-1}$. Vypalované jsou při nízkých teplotách (475 až 520 °C) po dobu 2 minut. Nízkou teplotou a krátkým časem výpalu se zabráňuje změnám elektrických parametrů základní vrstvy. Některá dielektrika s velkou permitivitou ε_r vyžadují ochrannou vrstvu z rekrystalizačního skla místo jednoduché skloviny [2]. Barevné skloviny se používají i pro značení.

1.4 Dostavování hodnot tlustovrstvých rezistorů

Tlustovrstvý rezistor je charakterizován parametry: plošný odpor R_{\square} , teplotní součinitel odporu TK_R , proudový šum, napěťový součinitel odporu, index zkreslení třetí harmonickou a stabilita odporu [2].

Celkový odpor R rezistoru je určen odporem na čtverec R_{\square} a počtem čtverců n (dle rovnice 1.5), který je závislý na rozměrech rezistoru. Hodnota odporu na čtverec závisí na použitém druhu odporové pasty a podle druhu pasty i na teplotě a době výpalu. Odpor na čtverec o stranách l a w lze vyjádřit rovnicemi 1.4, pokud je nám známa rezistivita materiálu ρ a tloušťka vrstvy h . Výchozí je rovnice 1.3, kde A představuje průřez a l délku.

$$R = \rho \frac{l}{A} \Omega \quad (1.3)$$

$$\begin{aligned} R_{\square} &= \rho \frac{l}{w \cdot h} \Omega \\ l &= w \end{aligned} \quad (1.4)$$

$$R_{\square} = \frac{\rho}{h} \Omega$$
$$R = R_{\square} \frac{l}{w} = R_{\square} n \Omega \quad (1.5)$$

Proudový šum A_1 závisí na rozměrech rezistoru. Se zmenšováním rozměrů rezistoru se zvětšuje proudový šum. Ke zmenšování rozměrů dochází například při dostavování rezistoru. Principem růstu A_1 je deformace proudových čar. U rezistoru kloubového tvaru lze dosáhnout při dostavování poklesu A_1 až do hraničního bodu. Po překročení hraničního bodu A_1 opět roste.

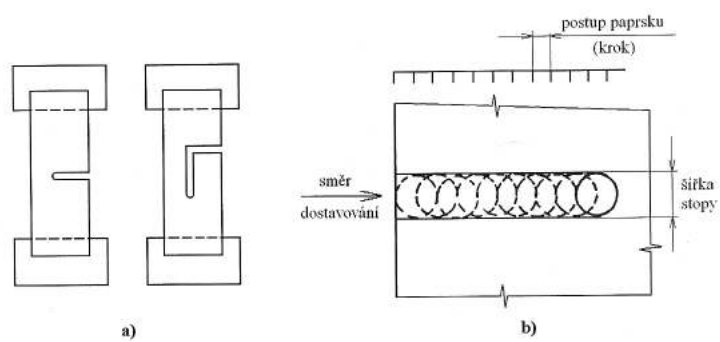
$$A_1 = 10 \log \frac{K_1}{lwh} \quad (1.6)$$

K_1 je materiálová konstanta

l, w, h jsou rozměry rezistoru (délka, šířka, tloušťka)

Vztah mezi napětím a proudem odporové vrstvy není zcela lineární podle Ohmova zákona, protože uvnitř struktury vznikají potenciálové bariéry vlivem fyzikálních a chemických procesů (difúze, rekrytalizace, reakce složek s okolím) [2]. Tuto nelinearitu popisuje napěťový součinitel, který se projevuje až při vysokých napětích.

Trimováním rezistoru lze zvyšovat hodnotu jeho odporu. Principem je zvětšení počtu čtverců rezistoru. Pro hrubé dostavení se používá zářez kolmý ke směru odporu a k jemnému dostavení zářez tvaru L, kdy je rezistor dostaven nejprve hrubě kolmým zářezem a následně jemně (obr. 1.4a). Pro dostavení trimovacím laserem lze použít kontinuální i pulzní režim (obr. 1.4b) paprsku. Přesnost dostavení hodnoty odporu může dosáhnout až 0.2 %.



Obr. 1.4: Princip trimování hodnoty rezistoru laserem [1]
a) dva různé tvary řezů b) znázornění odpařování vrstvy

2 VOLBA ZDROJOVÉHO FORMÁTU

Návrhové programy typu CAD jsou běžně používány pro kreslení technických výkresů. Výstupem těchto programů mohou být kromě jiných formáty PostScript, DWG či Gerber. Tyto formáty popisují 2D obrazec či 3D těleso. Pro řezání je žádán rovinný (2D) popis řezaného obrazce. Zdrojový formát je třeba převést do formátu, kterému rozumí laser Aurel ALS 300. Softwarové vybavení laseru (program PROTOMAT) umožňuje laseru řezat několik základních elementů, z nichž je podstatná schopnost řezat šikmou čáru. Z hlediska převodu je pak od optimálního zdrojového formátu očekáván přísun dat, která není potřebné zpracovávat složitými algoritmy.

2.1 PostScript

PostScript je programovací jazyk určený ke grafickému popisu tisknutelných dokumentů vyvinutý v roce 1985 firmou Adobe Systems Incorporated. Jeho hlavní výhodou je, že je nezávislý na zařízení, na kterém se má dokument tisknout. Je považován za standard pro dražší tiskárny. Díky svým rozsáhlým možnostem se však brzy stal i formátem k ukládání obrázků. [7]

Jedná se o interpretovaný jazyk, založený na práci se zásobníky. Jeho základními grafickými objekty jsou pak cesty, které lze obtahovat a vyplňovat, bitové mapy, rastry, vzorky a písma. Téměř celá jeho specifikace je volně dostupná na Internetu v publikaci PostScript Language Reference Manual (PLRM). [8]

Interpret PostScriptu postupně prochází kód, interpretuje jeho příkazy a provádí vykreslování (rastrování) do výstupní paměti. Interpretu se proto také někdy říká RIP (Raster Image Processor) a zmíněnému procesu pak slangově „ripování“. [8]

Písma, bitové mapy a rastry, jejichž výskyt je v PostScript dokumentu povolen, vyžadují vývoj algoritmu schopného vygenerovat řezy jdoucí podél vnějších obrysů obrázků či písmen. Při automatickém generování je zvláště u jednoduchých algoritmů pravděpodobný vznik značného množství řezů. Množství řezů představuje z hlediska řezání laserem zdržení, jelikož laser mezi každými dvěma řezy na okamžik zastaví, aby mohl přesunout hlavu na nové počáteční souřadnice.

Pro lepší představu o požadavcích na interpret PostScriptu je možné nahlédnout do ukázky PostScriptu (Listing 2.1). Pro úplnost je uveden také obrázek 2.1, který je popsán následující ukázkou PostScriptu.

Listing 2.1: Ukázka jednoduchého PostScriptu [8]

```
1  %!  
2  %nastaveni velikosti media  
3  1 dict dup /PageSize [40 40] put setpagedevice  
4  % definice zkratky pro prikaz curveto  
5  /c { curveto } def  
6  % uložíme na zásobník dva parametry  
7  % prikaz translate je vyzvedne a provede  
8  % nastaveni souradneho systemu  
9  -15.7606 -35 translate  
10 % nova cesta  
11 newpath  
12 % posun pera  
13 46.3294 35 moveto  
14 % nekolik Bezierovych krivek:
```

```

15 % curveto odebira ze zasobniku sest parametru:
16 % x1 y1 x2 y2 x3 y3
17 % pero se presune do [x3,y3]
18 % s pouzitim ridicich bodu [x1,y1] a [x2,y2]
19 29.2607 39.8018 16.8406 59.5148 15.7606 65.2856 c
20 17.6206 70.885 22.9607 73.9362 25.5407 74.355 c
21 27.5207 73.0986 31.7207 67.0067 32.5607 64.2173 c
22 31.3007 62.3182 29.0207 61.0105 27.4007 60.3524 c
23 27.8207 58.6772 35.5594 48.1696 41.1854 46.6044 c
24 41.7839 48.2297 43.1597 50.5775 45.0735 51.8417 c
25 47.8843 51.0591 53.9883 46.8451 55.305 44.9188 c
26 54.9459 42.3302 51.8936 36.9724 46.3294 35 c
27 % uzavreni cesty
28 closepath
29 % vyplneni oblasti ohranicene cestou
30 fill
31 % vykresleni stranky
32 showpage

```



Obr. 2.1: Výsledek vykreslení příkladu PostScriptu [8]

2.2 DWG

DWG je nativním formátem souborů (výkresů) programu AutoCAD. Název zdřejmě odvozen z DraWinG. Umožňuje ukládat 2D i 3D data. Díky rozšíření programu AutoCAD je tento formát spolu se svou výměnnou (textovou) variantou DXF považován za standard v oblasti výměny 2D CAD dat. Některé konkurenční programy dnes umožňují jeho čtení i zápis. Protože je to ale neveřejný formát firmy Autodesk, kompatibilita takto vytvořených souborů je neúplná a není zaručena. [9]

Na rozdíl od DXF, jehož specifikaci lze nalézt ve zdroji [10], je DWG formát binární. Binární uložení dat poskytuje jednoznačnou výhodu z hlediska velikosti souboru. Analýza binárních dat však oproti analýze dat textových skrývá mnoho problémů, které v textových formátech není zapotřebí řešit. U binárního formátu DWG můžeme také očekávat vyšší rychlost zpracování.

DWG formát není zcela ucelený. Existuje mnoho verzí DWG formátu. Některé verze pochází od firmy Autodesk, jiné od společnosti Open Design Alliance [11]. Verze DWG formátu od Open Design Alliance je částečně kompatibilní s verzemi DWG formátu od firmy Autodesk. Tato rozmanitost, současně s nenalezením oficiální specifikace binárního formátu DWG od Autodesku, činí tento formát nepříliš vhodným pro použití jako formátu zdrojového.

V neprospěch DWG formátu hraje i přítomnost textu. Konverzní algoritmus by tak v praxi musel text ignorovat, nebo s využitím vhodného algoritmu nahradit řezy.

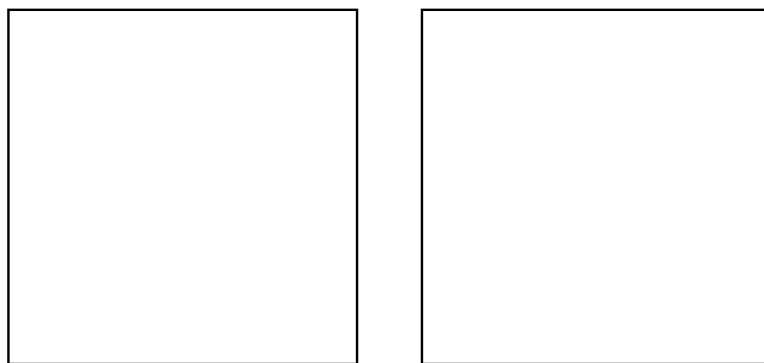
2.3 Gerber

Gerber formát, který je specifikován normou RS-274X [12], případně starší verzí [13], je často používaný ve výrobě desek plošných spojů (DPS). Formát je charakterizován svou jednoduchostí a textovým formátem dat. V Gerber formátu je využíván G-kód, který je určen k řízení NC a CNC obráběcích strojů (bližší informace viz [14]). Přestože je G-kód normován, jeho reálná podoba se liší dle výrobce. Tento problém se však netýká samotného Gerber formátu, v jehož normě jsou jednoznačně určeny podporované G-kódy a jejich význam.

V Gerber formátu není podpořen výskyt textu či obrázků. Není tak vyžadován vývoj algoritmů převádějící text či obrázky na řezy. Ukázka jednoduchého Gerber souboru (Listing 2.2) je doplněna znázorněním na obrázku 2.2.

Listing 2.2: Ukázka jednoduchého Gerber souboru [13]

```
1  *G04 EXAMPLE 1: 2 BOXES
2  %ASAXBY*%
3  %FSLAX23Y23*%
4  %MOIN*%
5  %OFA0B0*%
6  %SFA1.0B1.0*%
7  %ADD10C,0.010*%
8  %LNBOXES*%
9  G54D10
10 X0Y0D02*X5000Y0D01*
11 X5000Y5000D01*X0Y5000D01*X0Y0D01*
12 X6000Y0*X11000Y0D01*
13 X11000Y5000D01*X6000Y5000D01*
14 X6000Y0D01*D02*
15 M02*
```



Obr. 2.2: Gerber soubor převedený na obrázek [13]

3 KONVERZNÍ PROGRAM

V programu typu CAD vytvořený obrazec řezu je potřebné konvertovat z původního formátu do formátu, kterému rozumí program PROTOMAT, který řídí práci laseru. Tuto konverzi zajišťuje program GerberConversion.

Cílem programu je ze souborů zdrojového formátu generovat soubory ve formátu, kterému rozumí laser. Na základě analýzy formátů PostScript, DWG a Gerber je jednoznačné, že nejjednodušší je provést konverzi z Gerber formátu, který byl právě z tohoto důvodu zvolen jako výchozí. Je žádoucí, aby byl konverzní program schopen generovat i náhled znázorňující řezy. Z tohoto důvodu je program postaven na principu, který může vzdáleně připomínat tiskárnu. Program po spuštění dynamicky načte výstupní modul či moduly. Z načtených modulů (dynamických knihoven, v Microsoft Windows přípona .dll) jsou pak opakovaně volány funkce pro vykreslení řezu. Program je nazýván GerberConversion.

Součástí projektu je kromě programu GerberConversion a jeho modulů také program ViewALS300data, který však pouze zobrazuje informace, které obsahuje vstupní soubor laseru. Tento program byl vyvinut kvůli analýze vstupních dat laseru. K programu GerberConversion patří moduly Text, Bmp a AurelALS300data. Pro konverzi obrazce řezu do formátu srozumitelného programu PROTOMAT slouží modul AurelALS300data. Modul Bmp generuje náhledy řezů. Modul Text poskytuje informace potřebné pro ladění.

3.1 Charakteristika Gerber formátu

Na samém začátku Gerber souboru lze nalézt informace týkající se parametrů dle normy RS-274X. Jedná se o rotace, jednotky, formát souřadnic, měřítko, offset apod. Vzhledem k předpokladu generování souboru pro řezání lze některá z těchto nastavení ignorovat (polarita). Znak * v souboru odděluje datové bloky. Znak % pak uvozuje a ukončuje parametry normy RS-274X.

Vlastní řetězec vykreslování je řízen kódy D a G.

3.1.1 RS-274X parametry

Rozdělení parametrů dle normy [12]:

1. **Parametry určující způsob zpracování souboru.**
2. **Parametry obrazu** popisují obraz jako celek.
3. **Parametry nástrojů** přiřazují nástroje D kódům.
4. **Parametry vrstvy** popisují zpracování jedné nebo více vrstev.
5. **Různé parametry** poskytují možnosti, které nezapadají do výše uvedených skupin.

Parametry určující způsob zpracování souboru (Directive parameters)

Jedná se o příkazy ovlivňující zpracování souboru. Tedy například způsob čtení souřadnic a jiných číselných údajů ze zdrojového Gerber souboru. Přehled parametrů je v tabulce 3.1.

Tab. 3.1: Parametry určující způsob zpracování souboru [13]

AS	Axis Select	Výběr osy
FS	Format Statement	Formátování vstupů
MI	Mirror Image	Zrcadlení
MO	Mode of units	Jednotky
OF	Offset	Offset
SF	Scale Factor	Měřítko

Parametry obrazu (Image parameters)

S pomocí parametrů obrazu lze snadno dosáhnout například rotace. Některé parametry z této skupiny, například Image Polarity, však nemají pro potřeby generování řezů upotřebení. Přehled parametrů obrazu je v tabulce 3.2.

Tab. 3.2: Parametry obrazu [13]

IJ	Image Justify	Poloha obrazu
IN	Image Name	Jméno obrazu
IO	Image Offset	Offset obrazu
IP	Image Polarity	Polarita obrazu
IR	Image Rotation	Rotace obrazu
PF	Plotter Film	Označení filmu

Parametry nástrojů (Aperture parameters)

Parametry nástrojů jsou nástroje, pomocí kterých lze k různým D-kódům přiřadit vlastní kreslicí funkce. Přehled parametrů nástrojů je v tabulce 3.3.

Tab. 3.3: Parametry nástrojů [13]

AD	Aperture Description	Popis nástroje (clony)
AM	Aperture Macro	Makro nástroje (clony)

Parametry vrstvy (Layer-specific parameters)

Parametry vrstvy lze z většiny ignorovat, pokud jde o konverzní program Gerber formátů mající generovat řezy. Zajímavým je však parametr Step and Repeat, který definuje počet opakování vrstvy. Seznam parametrů vrstvy je v tabulce 3.4.

Tab. 3.4: Parametry vrstvy [13]

KO	Knockout	Knokaut, oblast opačné polarity
LN	Layer Name	Jméno vrstvy
LP	Layer Polarity	Polarita vrstvy
SR	Step and Repeat	Opakování vrstvy

Různé parametry (Miscellaneous parameters)

Mezi různé parametry se řadí jen parametr Include File, který slouží k vložení souboru. Parametr je uveden v tabulce 3.5.

Tab. 3.5: Různé parametry [13]

IF	Include File	Vložení souboru
----	--------------	-----------------

3.2 Analýza vstupních dat laseru

Ačkoli se na první pohled může zdát, že data pro laser jsou vždy tvořena dvojicí souborů (*.dat a *.txt), po nahlédnutí do zdrojových kódů (Listing 3.1) lze snadno dokázat, že rozhodující je soubor *.dat (dále datový soubor). Soubor *.txt (dále příkazový soubor) je volitelným doplňkem, který je hojně využíván k definici přesného postupu práce laseru.

Listing 3.1: Část funkce fileopener - načítání příkazového souboru [15]

```
1 thefn$ = thepath1$ + fn$ + ".dat"
2 thefn1$ = thepath1$ + fn$ + ".txt"
3 ex$ = ""
4 ex$ = Dir$(thefn1$)
5 If ex$ <> "" Then
6     un% = FreeFile
7     Open thefn1$ For Input As #un%
8     ll& = LOF(un%)
9     psistring$ = Input$(ll&, un%)
10    Close #un%
11 End If
12 ex$ = ""
```

3.2.1 Datový soubor

Datový soubor je tvořen hlavičkou o 37 řádcích. Za hlavičkou následují data určující jednotlivé řezy laseru. Data jsou tvořena 40 řádky a jsou ukládána s pevně danou velikostí.

Hlavička

Jak již bylo uvedeno, hlavička je tvořena 37 řádky. Nejpodstatnějšími informacemi obsaženými v hlavičce jsou reference, jak lze zjistit z části zdrojového kódu funkce fileopener (Listing 3.2). Z této funkce je také patrné, že hlavička je načítána po řádcích.

Listing 3.2: Část funkce fileopener - načítání hlavičky datového souboru[15]

```
1 ex$ = Dir$(thefn$)
2 If ex$ = "" Then lastfile$ = "\":_Exit_Sub
3 un%=_FreeFile
4 Open_thefn$ _For _Input _As _#un%
```

```

5 Line _Input _#un%,_LaserVersion$
6 Line _Input _#un%,_FileDescription$
7 Line _Input _#un%,_LasReferenceX$
8 Line _Input _#un%,_LasReferenceY$
9 Line _Input _#un%,_LasReferenceX1$
10 Line _Input _#un%,_LasReferenceY1$
11 Line _Input _#un%,_TabReferenceZ$
12 Line _Input _#un%,_TabReferenceW$
13 Line _Input _#un%,_TabStepZ$
14 Line _Input _#un%,_TabStepW$
15 Line _Input _#un%,_NumStepZ$
16 Line _Input _#un%,_NumStepW$
17 Line _Input _#un%,_TrimExe$
18 Line _Input _#un%,_usepsi$
19 Line _Input _#un%,_RRef1FocusX
20 Line _Input _#un%,_RRef1FocusY
21 Line _Input _#un%,_RRef1Ang
22 Line _Input _#un%,_RRef2FocusX
23 Line _Input _#un%,_RRef2FocusY
24 Line _Input _#un%,_RRef2Ang
25 Line _Input _#un%,_RRef3FocusX
26 Line _Input _#un%,_RRef3FocusY
27 Line _Input _#un%,_RRef3Ang
28 Line _Input _#un%,_RCalFocusPos
29 Line _Input _#un%,_RCalFocusANG
30 Line _Input _#un%,_LoadTmpString
31 MasterRefXProbeA _=_ Val ( LoadTmpString )
32 Line _Input _#un%,_LoadTmpString
33 MasterRefYProbeA _=_ Val ( LoadTmpString )
34 Line _Input _#un%,_LoadTmpString
35 MasterRefXProbeB _=_ Val ( LoadTmpString )
36 Line _Input _#un%,_LoadTmpString
37 MasterRefYProbeB _=_ Val ( LoadTmpString )
38 Line _Input _#un%,_LoadTmpString
39 CoordXFiducial1 _=_ Val ( LoadTmpString )
40 Line _Input _#un%,_LoadTmpString
41 CoordYFiducial1 _=_ Val ( LoadTmpString )
42 Line _Input _#un%,_LoadTmpString
43 CoordXFiducial2 _=_ Val ( LoadTmpString )

```



```

44 Line _Input _#un%, _LoadTmpString
45 CoordYFiducial2 _=_ Val ( LoadTmpString )
46 Line _Input _#un%, _strAsseStep
47 Line _Input _#un%, _strAsseOffset2
48 Line _Input _#un%, _strNCut
49 Line _Input _#un%, _VisionRefPt

```

S pomocí programu ViewALS300data, který je součástí této práce, lze zobrazit obsah jednotlivých datových souborů laseru. Nejprve je potřeba sledovat hlavičku a nalézt proměnné a konstantní hodnoty. Lze odpozorovat, že proměnné jsou hodnoty `LasReferenceX`, `LasReferenceY`, `TrimExe`, `MasterRefXProbeA`, `MasterRefYProbeA`, `MasterRefXProbeB` a `MasterRefYProbeB`. Hodnoty ostatních vlastností v hlavičce se ve zkoumaném vzorku datových souborů jeví jako konstantní.

LasReferenceX a **LasReferenceY** jsou referenčními body laseru. Jejich hodnotu lze chápat jako počátek souřadného systému. Poměrně optimální hodnota usnadňující ruční dostavení je 100 pro referenci v ose X i ose Y.

TrimExe je výpisem číselného pořadí elementů, které budou laserem vypáleny, pokud nebude použit příkazový soubor.

MasterRefXProbeA, **MasterRefYProbeA**, **MasterRefXProbeB** a **MasterRefYProbeB** jsou referenčními body určenými pro měřicí sondy. V případě řezání laserem ztrácí na významu. Užití nachází v případě trimování rezistoru laserem.

Data

Data následující za hlavičkou jsou tvořena 40 řádky. Na rozdíl od hlavičky je velikost každého řádku pevně dána. Načítání dat neprobíhá po řádcích jako v případě hlavičky, ale po blocích dané velikosti. Z těla funkce `getrecord` (Listing 3.3) je tato skutečnost patrná.

Listing 3.3: Funkce `getrecord` - načítání dat[15]

```

1 Sub getrecord ( fn%, rn%)
2   rrnn& = rn%
3   ll& = Len(myrecord)
4   thepos& = ((rrnn& - 1) * ll&) + filoffset&
5   Seek fn%, thepos&
6   thestring$ = Input$(ll&, fn%)
7   Call CopyMemory(myrecord, ByVal thestring$, ll&)
8 End Sub

```

Způsobu načítání odpovídá také definice typu trimrecord (Listing 3.4). Je nutné brát v úvahu skutečnost, že každý řádek končí dvojicí znaků v jazyce C zapsatelnou jako `"\r\n"`. Jde o sekvenci dvou bajtů nabývajících hodnot 0D a 0A v hexadecimálním zápisu. Část každého řádku nesoucí informaci pro laser je tedy o dva bajty menší než hodnota uvedená v definici.

Listing 3.4: Definice typu trimrecord [15]

```

1 Public Type trimrecord
2     'start x
3     startx As String * 13
4     'start y
5     starty As String * 13
6     'stop x
7     stopx As String * 13
8     'stop y
9     stopy As String * 13
10    'valore ohmico
11    'hodnota odporu
12    ohmvalue As String * 11
13    'velocita ' taglio
14    'rychlost řezu
15    speed As String * 7
16    'corrente
17    'proud
18    current As String * 7
19    'freq. laser
20    'frekvence laseru
21    freq As String * 7
22    'scanner h
23    'skener h
24    scanh As String * 5
25    'scanner l
26    'skener l
27    scanl As String * 5
28    'scanner g
29    'skener g
30    scang As String * 5
31    'slope
32    slope As String * 3

```

33 *'tolleranza accettazione pretest*
34 'tolerance předběžné zkoušky
35 tolaccept As **String** * 5
36 *'tolleranza taglio finale*
37 'tolerance konečného řezu
38 tolcut As **String** * 5
39 *'tipo di taglio*
40 'typ řezu
41 cutType As **String** * 4
42 *'direzione inizio taglio*
43 'počáteční směr řezání
44 cutDir As **String** * 3
45 *'percentuale primo stop*
46 'procento pro první zastávku
47 cut1St As **String** * 5
48 *'percentuale prestop*
49 'procento pro další zastavení
50 cutPre As **String** * 5
51 *'valore in volt scatto funzionale*
52 'funkční hodnota po klepnutí
53 voltvalue As **String** * 11
54 *'mark condition*
55 'označení stavu
56 mark As **String** * 3
57 *'taglio funzionale*
58 'funkční vyjmutí
59 func As **String** * 3
60 *'step taglio serp*
61 'krok řezu
62 serpstep As **String** * 6
63 *'identificatore di res.*
64 'identifikátor
65 reslabel As **String** * 12
66 *'se 1 usa visione*
67 'pokud se používají vize
68 vision As **String** * 3
69 *'i seguenti parametri vengono*
70 *' utilizzati per il taglio multiplo*
71 ' parametry používané

```

72      ' při vícenásobném řezání
73      'start2 X
74      start2X As String * 13
75      'start2Y
76      start2Y As String * 13
77      'stop2X
78      stop2X As String * 13
79      'stop2Y
80      stop2Y As String * 13
81      'N. Cut Multipli
82      strNCutMultiple As String * 13
83      'sviluppi futuri
84      sparedata As String * 1816
85      'Posizione X Probe A
86      'pozice X sondy A
87      recProbeAX As String * 13
88      'Posizione Y Probe A
89      'pozice Y sondy A
90      recProbeAY As String * 13
91      'Posizione X Probe B
92      'pozice X sondy B
93      recProbeBX As String * 13
94      'Posizione Y Probe B
95      'pozice Y sondy B
96      recProbeBY As String * 13
97      'Stato Filtro 1
98      'stavový filtr 1
99      recFilter1 As String * 3
100     'Stato Filtro 2
101     'stavový filtr 2
102     recFilter2 As String * 3
103     'Stato Filtro 3
104     'stavový filtr 3
105     recFilter3 As String * 3
106     'Stato Filtro 4
107     'stavový filtr 4
108     recFilter4 As String * 3
109     'Stato Filtro PSI
110     'stavový filtr PSI

```

```

111      recFilterPSI As String * 3
112      'Abilitazione AutoFuoco
113      'povolení autozaměření
114      AutoFEnable As String * 3
115 End Type

```

Opět analýzou datových souborů lze zjistit proměnná a konstantní data. Hodnoty `startx`, `starty`, `stopx`, `stopy`, `speed`, `current`, `freq`, `cutType` a `reslabel` jsou proměnné rozhodující pro řezání rovné čáry.

startx, starty, stopx a stopy udávají počáteční a konečné souřadnice pro řezání.

speed, current a freq udávají vlastnosti paprsku laseru a rychlost jeho pohybu.

cutType udává typ řezu, který má být laserem proveden. Hodnoty odpovídající jednotlivým typům řezu lze dohledat ve zdrojovém kódu (Listing 3.5).

Listing 3.5: Typy řezu - funkce `From_Load` [15]

```

1  lsttype.Clear
2  lsttype.AddItem "Double"           '0
3  lsttype.AddItem "L"               '1
4  lsttype.AddItem "Mark_Circle"    '2
5  lsttype.AddItem "Mark_Square"    '3
6  lsttype.AddItem "Serp."          '4
7  lsttype.AddItem "Single"         '5
8  lsttype.AddItem "Ortho_uncon."   '6
9  lsttype.AddItem "Unconditional"  '7
10 lsttype.AddItem "Multiple_Cut"  '8

```

reslabel je pojmenováním řezu. Jméno má pouze informativní význam.

3.2.2 Příkazový soubor

Příkazový soubor obsahuje sekvenci jednoduchých příkazů, které řídí práci laseru. Detailní informace lze nalézt v manuálu laseru [16]. Při použití laseru k řezání si lze vystačit jen s několika základními příkazy.

,

Uvádí komentář, který končí na konci řádku.

1 'komentar

CLS

Vymaže obrazovku.

1 CLS

EQU

Přiřazení hodnoty do proměnné.

1 EQU C1 0

PRINT

Tiskne text a ukončuje řádek.

1 PRINT "Text"

#

Definuje místo, kam lze skočit. Místo pro skok je určeno číslem.

1 #900

D

Provede řez. Číslo řezu odpovídá pořadí řezu v datovém souboru počítáno od jedné.

1 D 1

SUM

Prezentuje součet. Parametry jsou proměnné pro uložení výsledku, první sčítanec a druhý sčítanec.

1 SUM C1 C1 1

2 'C1=C1+1

JLESS

Pokud je první parametr menší než druhý parametr, skoč na místo určené třetím parametrem.

```
1 JLESS C1 C2 900
2 'je-li C1 mensi C2, skoc na misto urcene zapisem #900
```

WAIT

Čeká stanovený počet milisekund.

```
1 WAIT 1000
```

GOTO

Nepodmíněný skok.

```
1 GOTO 900
```

END

Konec příkazového souboru.

```
1 END
```

3.3 Generování výstupních dat

Výstupní data lze vytvářet za běhu a nebo až po přečtení všech požadovaných řezů. Oba způsoby mají své výhody a nevýhody. Moduly napsané pro program GerberConversion využívají obou způsobů.

3.3.1 Generování výstupních dat v průběhu čtení vstupního souboru

Tato metoda vyžaduje minimální paměťové nároky. Za určitých okolností stačí i minimální vývojové nároky. Tohoto způsobu užívají moduly Text a Bmp. V případě modulu Text je výpis za běhu velmi optimálním a silným řešením. V případě modulu Bmp není již situace natolik jednoznačná. Nejednoznačnost je dána komplikacemi při generování Bmp obrázku, které jsou způsobeny neznámou konečnou velikostí obrázku a neznámou konečnou pozicí obrázku vůči počátku souřadného systému. Na rozdíl od modulu Text je tedy v případě Bmp modulu použití tohoto způsobu spíše vývojovou komplikací.

3.3.2 Generování výstupních dat po dokončení čtení vstupního souboru

Z principu metody je vyžadováno provedení záznamu všech řezů. V průběhu záznamu je samozřejmě možné data vhodným způsobem zpracovat. Avšak předpokládá se zpracování všech řezů současně po dokončení záznamu. Tento princip je využíván modulem AurelALS300data, který přijatá data zpracovává a zaznamenává ve zpracované podobě, ze které následně generuje výstupní soubory. Před vlastní generováním výstupních souborů je navíc možné provést transformaci zaznamenaných řezů.

3.4 Charakteristika programu

Jak již bylo naznačeno, konverzní program je navržen jako modulární, ačkoli úplně původní myšlenka byla jednodušší. Myšlenka modulárního provedení se však ukázala být výhodná nejen z hlediska vývoje.

3.4.1 Nemodulární provedení

V případě nemodulárního systému by vznikl jeden programový celek, který by při potřebě přidání funkce či opravy chyby musel být vždy upravován. Tím by výsledný program nebyl flexibilní. Lze předpokládat i větší provázanost zdrojových kódů a s tím související složitější ladění programu.

3.4.2 Modulární provedení

Modulární provedení těží z relativní jednoduchosti aplikace analyzující Gerber soubor. Aplikace se pouze odvolává do modulu a informuje jej o řezech, které je nutné provést. Každý modul je realizován dynamickou knihovnou, která obsahuje několik jasně daných funkcí. Problém konverze je tak modulárním systémem rozdělen na dvě samostatné části. První část problému řeší vlastní program GerberConversion, který analyzuje vstupní Gerber soubor. Druhá část je řešena modulem, který se stará o vytvoření žádaného výstupu (například o vytvoření náhledu či vstupních dat pro laser). Díky modulu, který zajišťuje výstup, je modulární provedení flexibilní. Teoreticky stačí vyvinout nový modul, aby bylo možné generovat vstupní data pro jiný laser.

3.4.3 Modul

V modulu se musí vyskytovat povinně tři funkce. Jedna funkce je volitelná. Definici povinné funkce `fGerber_Module_Start` a nepovinné funkce `fGerber_Module_End` lze nalézt v hlavičkovém souboru `Module.h` (Listing 3.6). Funkce `fGerber_Module_Start` je volána vždy během načítání modulu. V případě úspěšného provedení vrátí 0. Parametr `pv_memory` představuje ukazatel na místo, kam lze uložit ukazatel, který bude následně předán každému volání funkce kreslící řez i funkci `fGerber_Module_End`.

Listing 3.6: Definice funkcí `fGerber_Module_Start` a `fGerber_Module_End`

```
1 typedef int (*fGerber_Module_Start)(const char **at_args ,  
2                                     uint32_t u4_args , void **pv_memory );  
3 typedef void (*fGerber_Module_End)(void *v_memory );
```

Pro moduly je důležitý také hlavičkový soubor `Draw.h` obsahující definice funkcí `fGerber_Draw_LineSegment` a `fGerber_Draw_CircleSegment` (Listing 3.7), které musí

být v modulu povinně. Tyto funkce jsou volány v průběhu analyzování Gerber souboru. Funkce `fGerber_Draw_CircleSegment` však ve verzi programu `GerberConversion`, který je výsledkem této semestrální práce, volána není. Její význam je v budoucím použití po přidání dalších funkcí programu `GerberConversion`.

Listing 3.7: Definice funkcí `fGerber_Draw_LineSegment` a `fGerber_Draw_CircleSegment`

```
1 struct sGerber_Point
2 {
3     long double ld_x , ld_y ;
4 };
5
6 typedef int (*fGerber_Draw_LineSegment)
7             (void *v_mem, struct sGerber_Point *ps_From ,
8             struct sGerber_Point *ps_To , bool b_positive );
9 typedef int (*fGerber_Draw_CircleSegment)
10            (void *v_mem, struct sGerber_Point *ps_centre ,
11            long double ld_radius , long double ld_aglefrom ,
12            long double ld_agleto , bool b_positive );
```

Aby bylo možné funkce v načítaném modulu, který je tvořen dynamickou knihovnou, nalézt, musí mít tyto funkce pevně daná jména. Těmito jmény jsou "Start", "End", "LineSegment" a "CircleSegment". Z názvů je patrné, který název se váže ke které funkci.

K modulům je kromě hlavičkových souborů `Module.h` a `Draw.h` nutné přidávat také hlavičkový soubor `Headers.h`, na který se odkazuje hlavičkový soubor `Draw.h`. `Headers.h` pouze inkluduje standardní knihovny jazyka C.

Modul Text

Modul `Text` se zabývá jen textovým výpisem informace charakterizující to, co je po modulu žádáno programem `GerberConversion`. Důvodem realizace tohoto modulu bylo ladění programu `GerberConversion`. Zde modul umožnil pohodlný náhled na hodnoty, které generoval program `GerberConversion`.

Modul Bmp

Účelem modulu `Bmp` je generování náhledů. Je nutné kompenzovat nevýhodu Gerber formátu spočívající v neznalosti konečné velikosti výkresu. Náhled je proto automaticky posouván co nejbližší k počátku a jeho velikost je udržována na malém rozměru. Velikost rozměru je určitým kompromisem mezi jednoduchostí a velikostí

potřebné paměti. Délky stran výsledného nákresu musí být násobkem 8. Důvodem je, že modul interně pracuje s černobílým rastrem, kde je každý pixel představován jedním bitem. Zarovnání na násobek 8 je tak zarovnáním na celé bajty. To usnadňuje práci s grafickými daty.

Modul Bmp je nutné před použitím patřičně nastavit. Konkrétní dostupná nastavení můžeme nalézt v tabulce 3.6.

Tab. 3.6: Parametry modulu Bmp

Parametr	Příklad	Defaultní hodnota	Nutný?	Význam
--bmp =filename	--bmp =obr.bmp	NULL	ano	jméno vytvořeného souboru
--x_scaler =scaler	--x_scaler =10.0	1.0	ne	měřítko v ose x
--y_scaler =scaler	--y_scaler =10.0	1.0	ne	měřítko v ose y
--debug =level	--debug =3	0	ne	ovlivňuje podrobnost debugovacích výpisů

Modul AurelALS300data

Modul AurelALS300data zajišťuje generování samotných dat pro laser. Jako jediný generuje výstup až poté, co dostane informace o všech řezech. Tento model je využíván kvůli vlastnostem transformace a kvůli generování hlavičky datového souboru (vlastnost TrimExe).

Modul je schopen generovat datový i příkazový soubor. Příkazový soubor je generován v podobě, která umožňuje snadno měnit počet řezů. V příkazovém souboru jde o osmý řádek, kde se přiřazuje hodnota proměnné C2. Konkrétně se jedná o kód EQU C2 x, kde x představuje počet řezů.

Modul AurelALS300data je vhodné před použitím patřičně nastavit. Konkrétní dostupná nastavení je možné nalézt v tabulce 3.7.

Tab. 3.7: Parametry modulu AurelALS300data

Parametr	Příklad	Defaultní hodnota	Význam
--data =filename	--data =out.dat	NULL	jméno vytvořeného datového souboru
--commands =filename	--commands =out.txt	NULL	jméno vytvořeného příkazového souboru
--LaserReferenceX =value	--LaserReferenceX =10.0	100.0	souřadnice x referenčního bodu laseru
--LaserReferenceY =value	--LaserReferenceY =10.0	100.0	souřadnice y referenčního bodu laseru
--ProbeAReferenceX =value	--ProbeAReferenceX =1.0	0.0	souřadnice x referenčního bodu měřící sondy A
--ProbeAReferenceY =value	--ProbeAReferenceY =1.0	0.0	souřadnice y referenčního bodu měřící sondy A
--ProbeBReferenceX =value	--ProbeBReferenceX =1.0	0.0	souřadnice x referenčního bodu měřící sondy B
--ProbeBReferenceY =value	--ProbeBReferenceY =1.0	0.0	souřadnice y referenčního bodu měřící sondy B
--width =value	--width =10.0	50.0	šíře plochy pro transformaci
--height =value	--height =10.0	50.0	výška plochy pro transformaci

--current =value	--current =10.0	15.0	proud tekoucí laserem [A], ovlivňuje výkon laseru
--frequency =value	--frequency =2	1	frekvence laseru v kHz
--speed =value	--speed =1.0	0.5	rychlost posunu laserového paprsku [mm.s ⁻¹]
--cuts =level	--cuts =1	10	počet opakování řezu
--transformate	--transformate	zakázána	povoluje transformaci
--transformatetoarea	--transformatetoarea	zakázána	transformace do oblasti
--transformatetozero	--transformatetozero	zakázána	transformace do nuly
--debug =level	--debug =3	0	ovlivňuje podrobnost debugovacích výpisů

3.4.4 Rozsah podpory Gerber formátů

Podpora Gerber formátu v programu GerberConversion vytvořeného v rámci tohoto projektu se omezuje na nejběžnější kódy a parametry. Pro praktické použití programu GerberConversion je proto velmi důležité aktivovat ignoraci špatných kódů a parametrů. Z hlediska kódu jsou prozatím podporovány pouze rovné čáry, které však pokrývají většinu požadovaných případů reálného použití laseru.

3.4.5 Parametry

Prvním parametrem programu je cesta k vlastnímu spustitelnému souboru. Další parametry jsou zpracovány kódem, který předpokládá, že prvním následujícím parametrem je jméno konvertovaného Gerber souboru. Po parametru s názvem konvertovaného Gerber souboru se předpokládá výskyt parametrů (tabulka 3.8) vztahujících

se k vlastnímu programu GerberConversion a následně výskyt jmen modulů následovaných jejich parametry. Detekce parametrů probíhá na základě prvního znaku, kterým musí být pomlčka '-'. Za parametrem může následovat další parametr, který obsahuje údaj vztahující se k předchozímu parametru.

Cyklus analyzující parametry nejprve hledá parametry programu GerberConversion. Pokud prvním znakem není pomlčka, je tento parametr považován za jméno modulu. Touto podmínkou ale neprochází text, který je například parametrem parametru – transformsets. Pro bližší informace si můžete prohlédnout soubor main.c, ve kterém se kód uvedeného cyklu nachází. Po detekci jména modulu je již situace o něco obtížnější. Pro každý modul je nutné detekovat rozsah jeho parametrů a s těmi modul načíst. Za parametr modulu je považován každý parametr za jménem modulu, který začíná znakem pomlčky '-' nebo rovnítko '='. První parametr, který tuto podmínku nesplňuje, je považován za název dalšího modulu. Počet načítaných modulů není z principu omezen. Paměť je alokována dynamicky.

Tab. 3.8: Parametry GerberConvertoru

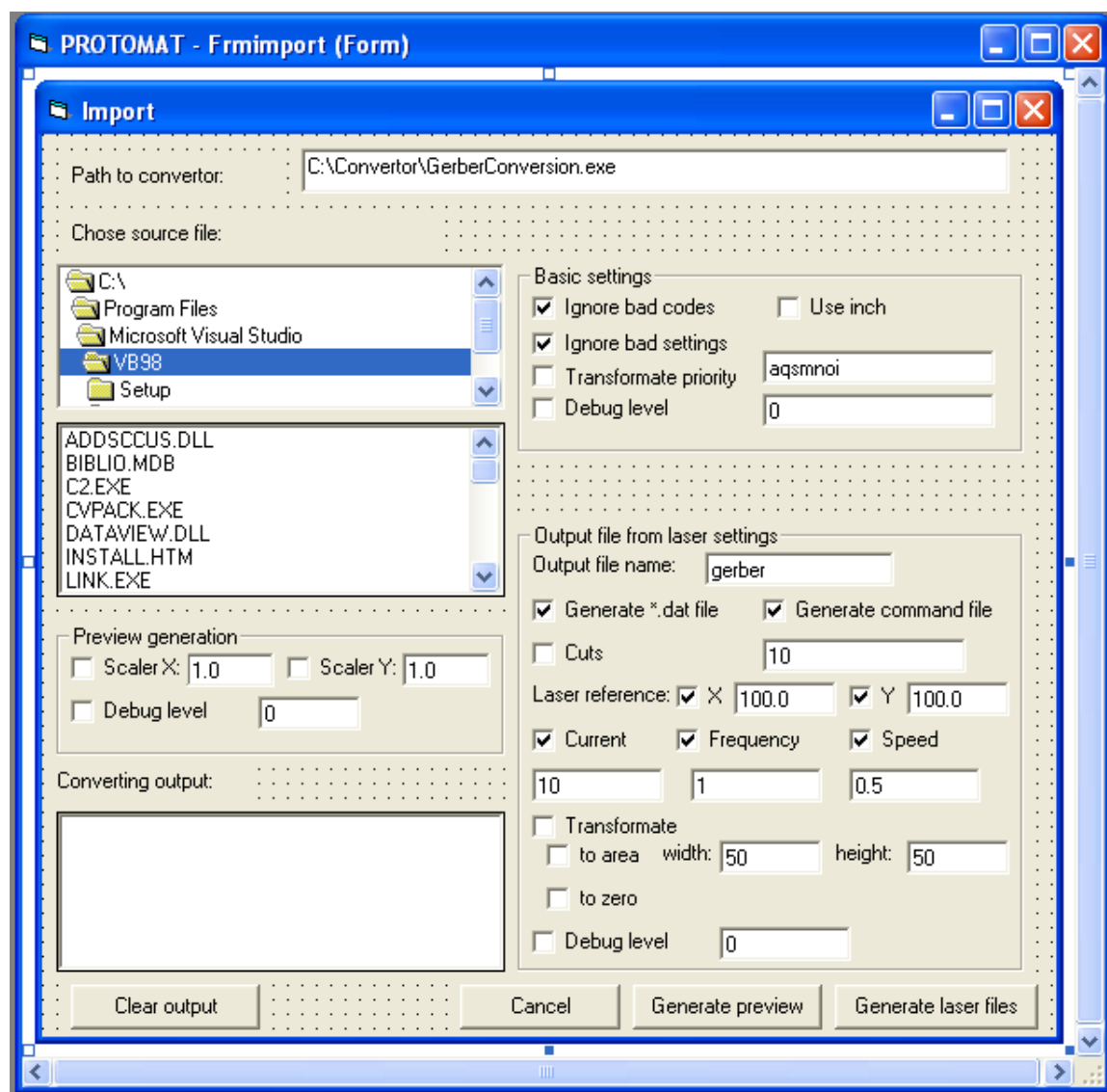
Parametr	Příklad	Defaultní hodnota	Význam
--ignore_badcodes	--ignore_badcodes	neignorováno	zapnutí ignorace špatných kódů
--ignore_badsets	--ignore_badsets	neignorováno	zapnutí ignorace špatných parametrů Gerberu
--inch	--inch	milimetry	přepnutí na jednotky inch
--transformsets text	--transformsets aqsmnoi	aqsmnoi	defnuje postup použitý při transformaci zadaných souřadnic z Gerber souboru do souřadnic konečných, určujících řez

3.5 Integrace

Z hlediska použitelnosti není praktické nutit uživatele pouštět konverzní program z příkazové řádky. Do původního programu laseru PROTOMAT bylo proto přidáno okno (obrázek 3.1), které konverzi usnadňuje. Toto okno lze vyvolat z menu File přes položku Import. V pravé horní části okna lze vybrat soubor určený k převodu. Pod výběrem souboru lze nalézt nastavení ovlivňující generování náhledu. Do textového okna vlevo dole je vypisován výstup konverzního programu. Vpravo nahoře lze ovlivnit základní nastavení programu GerberConversion a níže pak nastavení modulu AurelALS300data. Úplně nahoře je textové pole umožňující měnit cestu ke konverznímu programu, který bude použit.

Pro generování náhledu slouží tlačítko s nápisem "Generate preview". Po stisku tohoto tlačítka je vygenerován náhled do souboru preview.bmp, který je následně automaticky otevřen a zobrazen. Tlačítkem "Generate laser files" lze generovat vstupní soubory laseru. Po dokončení generování se program zeptá, zda si přejete generované soubory otevřít. Před otevřením je dobré zkontrolovat výpis konverzního programu. V průběhu konverze může dojít k chybě například v důsledku poškozeného vstupního souboru. Tlačítkem "Cancel" lze celé okno opustit.

Pokud je požadováno importování více souborů, je třeba pokaždé změnit obsah kolonky "Output file name". Program nekontroluje, zda daný soubor již existuje. Pokud existuje, dojde k jeho přepsání. Tlačítko "Clear output" slouží pouze k promazání zobrazeného výstupu.



Obr. 3.1: Návrh okna importu Gerber souboru do programu ovládajícího laser ve Visual Basicu 6

4 ZÁVĚR

V rámci této práce byly napsány programy GerberConversion a ViewALS300data. Program GerberConversion obstarává s pomocí modulů konverzi souborů formátu Gerber do požadovaného formátu. K programu GerberConversion byla dále vytvořena trojice modulů. Vytvořenými moduly jsou Text, Bmp a AurelALS300data. Program ViewALS300data zobrazuje data, která jsou obsažena v datovém souboru laseru.

V rámci bakalářské práce bude v programu GerberConversion rozšířena podpora Gerber formátu, a to nejlépe do úrovně plně podporující normu RS-274X. Další možnosti rozšiřování se vztahují přímo k programu PROTOMAT. V programu PROTOMAT je možné jednak zlepšit integraci GerberConvertoru a dále přichází v úvahu také zlepšení možností řezání.

LITERATURA

- [1] GRUND, Pavel. *Nekonvenční aplikace hybridních integrovaných obvodů* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010, 43 s. Vedoucí bakalářské práce Doc. Ing. Ivan Szendiuch, CSc.
- [2] SOUTOR, Zdeněk - ŠAVEL, Josef - ŽŮREK, Jaroslav. *Hybridní integrované obvody* 1. vyd., Praha: nakladatelství SNTL - Nakladatelství technické literatury, 1982. 400 s. ISBN 04-532-8
- [3] HC electronics, s. r. o. *Tlustovrstvá hybridní technologie* [online]. [cit. 17. 11. 2010]. Dostupné z URL: <<http://www.hcelectronics.cz/hiotlust.php>>.
- [4] SZENDIVCH, Ivan, a jiní. *Technologie elektronických obvodů a systémů* 1. vyd., Brno: nakladatelství VUTIUM, 2002. 239 s. ISBN 80-214-2072-3
- [5] STARÝ, JIŘI. *PLOŠNÉ SPOJE A POVRCHOVÁ MONTÁŽ, BPSM, 2. část Aplikální technologie (tisk a dispenze), Osazování, Pájení, opravy,, čištění, Kontrola kvality* [online]. pracovní verze 91220, 2010?, [cit. 26. 11. 2010]. Dostupné z URL: <https://www.vutbr.cz/www_base/priloha.php?dpid=25509>.
- [6] VAŠKO, Cyril, a jiní. *Vrstvové technologie* [online]. 2007, [cit. 2. 12. 2010]. Dostupné z URL: <<http://www.umel.feec.vutbr.cz/~vasko/multimedialni-ucebnice/vrstvove-technologie/>>.
- [7] *PostScript* [online]. poslední aktualizace 7. 9. 2010 [cit. 17. 11. 2010]. Dostupné z URL: <<http://cs.wikipedia.org/wiki/PostScript>>.
- [8] Brabec, Stanislav *Grafika v UNIXu - PostScript* [online]. poslední aktualizace 15. 10. 2001 [cit. 17. 11. 2010]. Dostupné z URL: <<http://www.root.cz/clanky/grafika-v-unixu-ix-postscript/>>.
- [9] *DWG* [online]. poslední aktualizace 29. 10. 2010 [cit. 18. 11. 2010]. Dostupné z URL: <<http://cs.wikipedia.org/wiki/DWG>>.
- [10] Autodesk. *DXF Reference* [online]. San Rafael, USA, 2010 [cit. 18. 11. 2010]. Dostupné z URL: <http://images.autodesk.com/adsk/files/acad_dxf2.pdf>.
- [11] Open Design Alliance. *Open Design Specification for .dwg files* [online]. Ver. 5.1. Phoenix, USA, 2010 [cit. 18. 11. 2010]. Dostupné z URL: <http://opendesign.com/files/guestdownloads/OpenDesign_Specification_for_.dwg_files.pdf>.

- [12] Autodesk. *Gerber RS-274X Format User's Guide* [online]. Gent, Belgium, 1998, Revision F, February 2010 [cit. 18. 11. 2010]. Dostupné z URL: <<http://www.ucamco.com/downloads.php>>.
- [13] Autodesk. *Gerber RS-274X Format User's Guide* [online]. Gent, Belgium, 1998, Revision D, March 2001 [cit. 18. 11. 2010]. Dostupné z URL: <http://www.artwork.com/gerber/274x/rs274xrevd_e.pdf>.
- [14] *G-kód* [online]. poslední aktualizace 30. 5. 2010 [cit. 18. 11. 2010]. Dostupné z URL: <<http://cs.wikipedia.org/wiki/G-kód>>.
- [15] AUREL AUTOMATION S.p.A. *PROTOMAT* [CD-ROM]. Modigliana, Italy, 17.10.2006 [cit. 18. 11. 2010]. Adresář: C:\Laser\
- [16] AUREL AUTOMATION S.p.A. *OWNER'S MANUAL: ALS300L LASER TRIMMING SYSTEM* [CD-ROM]. Ver. 1.0. Modigliana, Italy, 17.10.2006 [cit. 18. 11. 2010]. Serial no.: 3712099029

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

IO Integrovaný obvod

HIO Hybridní integrovaný obvod

DPS Deska plošného spoje